

# Stacking from Tags: Clustering Bookmarks around a Theme

Arkaitz Zubiaga  
Queens College and Graduate Center  
City University of New York  
New York, NY, USA  
arkaitz@zubiaga.org

Alberto Pérez García-Plaza,  
Víctor Fresno, Raquel Martínez  
NLP & IR Group  
ETSI Informática UNED  
Madrid, Spain  
{alpgarcia, vfresno, raquel}@lsi.uned.es

## ABSTRACT

Since very recently, users on the social bookmarking service Delicious can stack web pages in addition to tagging them. Stacking enables users to group web pages around specific themes with the aim of recommending to others. However, users still stack a small subset of what they tag, and thus many web pages remain unstacked. This paper presents early research towards automatically clustering web pages from tags to find stacks and extend recommendations.

## 1. INTRODUCTION

Social tagging has become a powerful means to organize resources facilitating later access [1]. On these sites, users can label web pages with tags, facilitating future access to those web pages both to the author and to other interested users [2, 7]. Very recently, Delicious.com introduced a new dimension for organizing web pages: stacking. With stacks, users can group the web pages that might be of interest for a specific community, e.g., *Valentine's Special*, or *UNIX and Programming Jokes*. Stacks can be very useful for those who are looking for help or information on a specific matter. With stacks, users are providing a 2-dimensional organization of web pages that is complemented with tags, as shown by the example in Table 1. However, tagging activity still clearly exceeds the stacking activity, and many web pages are tagged but not stacked. Moreover, all the web pages tagged before the new feature was introduced are not associated with stacks. Thus, finding a way to infer missing stacks from all those tags would be helpful to recommend more groups of web pages to communities, or to suggest adding web pages to any user stack. Different from previous research on clustering and classifying tagged resources, which evaluated using the Open Directory Project [4, 5] or from manually built categorizations [6], stacks provide a rather ad hoc ground truth to evaluate with. This paper describes early research in this direction, presenting preliminary work on automatically clustering web pages from tags

Stack #1			Stack #2			
URL <sub>1</sub>	URL <sub>2</sub>	URL <sub>3</sub>	URL <sub>4</sub>	URL <sub>5</sub>	URL <sub>6</sub>	URL <sub>7</sub>
tags <sub>1</sub>	tags <sub>2</sub>	tags <sub>3</sub>	tags <sub>4</sub>	tags <sub>5</sub>	tags <sub>6</sub>	tags <sub>7</sub>

Table 1: Example of a user's tags and stacks. The user tagged 7 URLs, with 5 of them in 2 stacks.

to find stacks as users would do. Preliminary experiments suggest that using tags can help reach high performance clustering.

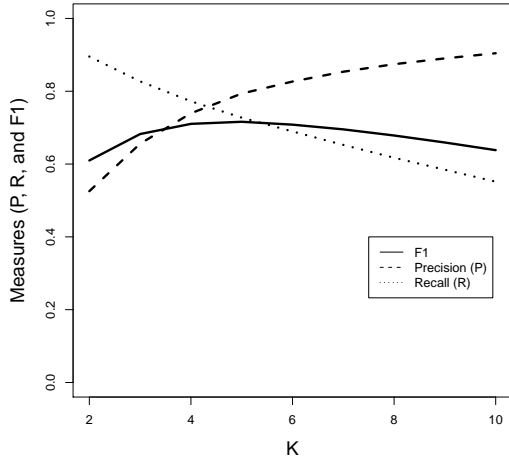
## 2. DATASET

We collected the tagging activity for 3,635 Delicious users in October and November 2011. This subset includes all the users who created at least a stack in this timeframe. During this period, those users tagged 182,510 web pages, creating 5,214 stacks. Out of those web pages, 45,196 (24.8%) were stacked while 137,314 (75.2%) were left out of stacks. Also, a large set of users who are not included in our dataset are tagging web pages, while they are not creating stacks. Going into further details on the tagging activity in and out of the stacks, we observe that, on average, 30.1% of the tags contained in stacks are also used out of the stacks. This suggests that there is not specific vocabulary for stacks, but users share vocabulary with web pages out of stacks. Moreover, there is just a small subset of 22.5% of the stacks that have a common tag in all their underlying web pages. Hence, most users do not use an exclusive tag that refers to the stack. This motivates our study on the automatic clustering of web pages from tags with the aim of finding stacks that approach to those created by users.

## 3. EXPERIMENTS

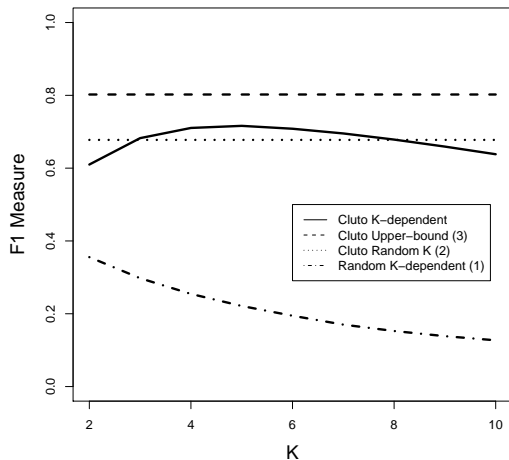
We used *Cluto rbr* [3] (k-way repeated bisections globally optimized) to find clusters from tags. *Cluto rbr* conveniently fits with the present task since, in practice, it always generates the same clustering solution for a certain input data. As the main parameter, this algorithm requires as an input the number of clusters to generate, which is known as  $K$ . We used values ranging from 2 to 10 for  $K$ , as a preliminary approach that allows us to evaluate and understand how the number of created clusters affects the solution. We set the rest of the parameters to their default values. Upon these settings, we got the resulting clusters for all the web pages saved by each user, and compare the results to the

stack(s) created by the user. For each run on a stack, we computed the precision, recall and F1 values, and got the macroaveraged values for all the stacks.



**Figure 1: Macroaveraged F1, P and R for K-dependent Cluto runs, ranging from 2 to 10 for K.**

Figure 1 shows precision, recall, and F1 values for K-dependent Cluto runs. Precision and recall values considerably vary depending on the selected  $K$  value. Creating a few big clusters improves recall, while creating many small clusters improves precision. However, F1 values remain very similar while  $K$  changes. Regardless of the value selected for  $K$ , the clustering gets F1 above 0.6. Hence, the selection of the value for  $K$  mainly conditions that the results get affected by either precision or recall, depending on the preference.



**Figure 2: Comparison of F1 values for K-dependent Cluto runs as compared to benchmark approaches.**

Figure 2 complements the above results by showing the F1

values for our K-dependent runs as compared to 3 benchmark methods: (1) a baseline approach that randomly creates the clusters, i.e., randomly generating  $K$  clusters of equal size, (2) an intermediate approach that randomly selects the value of  $K$  for each user, i.e., the average of multiple runs using random  $K$  values, and (3) the ideal upper-bound performance by choosing the optimal  $K$  value for each user stack. These results show that using tags to find stacks clearly outperforms a random approach, doubling the performance in many cases. This encourages the use of tags to perform this task in an effective way. Moreover, even though this paper does not explore how to find a suitable  $K$  for each stack, the upper-bound performance based on optimal  $K$  values shows that tags can reach very high results. An appropriate selection of  $K$  could yield clusters approaching to 0.8 performance in terms of F1. It also clearly outperforms the random selection of  $K$ , encouraging to perform further research in a way of looking for a suitable  $K$  for each user.

## 4. CONCLUSIONS

This work describes early research for a work-in-progress on a novel feature of social bookmarking systems: stacking. To the best of our knowledge, this is the first research work that deals with stacks. We have shown that the use of tags to find stacks that resemble to those created by users scores high performance results above 0.6 in terms of F1. Moreover, choosing the right parameters for each stack to be created can substantially improve performance by scoring nearly 0.8. As a preliminary work, these results encourage performing further study that helps make a decision on the selection of parameters that improves performance. Future work includes studying behavioral patterns of users such as tagging vocabulary towards finding the right parameters for each user. The promising results by using tags to discover stacks also suggest further research looking for groups of related tags both to individual users and communities.

## 5. REFERENCES

- [1] S. Golder and B. Huberman. Usage patterns of collaborative tagging systems. *Journal of information science*, 32(2):198–208, 2006.
- [2] P. Heymann, G. Koutrika, and H. Garcia-Molina. Can social bookmarking improve web search? In *Proceedings of WSDM 2008*, pages 195–206. ACM, 2008.
- [3] G. Karypis. CLUTO - a clustering toolkit. Technical Report #02-017, Nov. 2003.
- [4] D. Ramage, P. Heymann, C. Manning, and H. Garcia-Molina. Clustering the tagged web. In *WSDM*, pages 54–63, 2009.
- [5] A. Zubiaga, V. Fresno, R. Martínez, and A. P. García-Plaza. Harnessing folksonomies to produce a social classification of resources. *IEEE TKDE*, 99(Preliminary), 2012.
- [6] A. Zubiaga, A. P. García-Plaza, V. Fresno, and R. Martínez. Content-based clustering for tag cloud visualization. In *ASONAM*, pages 316–319, 2009.
- [7] A. Zubiaga, R. Martínez, and V. Fresno. Getting the most out of social annotations for web page classification. In *DocEng*, pages 74–83, 2009.